Supreme Court, U.S.
FILED
FEB 26 1998
CLERK

No. 97-999

IN THE

Supreme Court Of The United States

OCTOBER TERM, 1997

SOFTEL, INC.,

Petitioner,

– v. –

DRAGON MEDICAL AND SCIENTIFIC COMMUNICATIONS, INC.; DRAGON GROUP LTD., also known as Dragon Medical and Scientific Communications, Ltd.; JOHN R. DARSEE, H. EUGENE HODGE, NINA ROMANOFF,

Respondents.

ON PETITION FOR A WRIT OF CERTIORARI TO THE UNITED STATES COURT OF APPEALS FOR THE SECOND CIRCUIT

PETITIONER'S REPLY BRIEF

CHARLA R. BIKMAN
Counsel for Petitioner
25 Maple Lane
East Hampton, N.Y. 11937
(516) 324-1306

TABLE OF CONTENTS

TABLE OF CONTENTS i

TABLE OF AUTHORITIES i

I. THE ISSUES PRESENTED IN THE PETITION FOR A WRIT OF CERTIORARI WEE PRESSED OR PASSED UPON BY THE COURT OF APPALS AND ARE RIPE FOR REVIEW 1

II. THIS CASE CONCERNS ISSUES OF FUNDAMENTAL IMPORTANCE TO THE SOFTWARE INDUSTRY 4

CONCLUSION 8

TABLE OF AUTHORITIES

Cases Cited

Bateman v. Mnemonics, Inc., 79 F.3d 1532 (11th Cir. 1996) 7

Computer Associates Intern., Inc. v. Altai, Inc., 982 F.2d 693 (2nd Cir. 1992) 2, 5, 6, 8

Feist Publications, Inc. v. Rural Tel. Service, Co., 499 U.S. 340 (1991) 2, 3, 8

Harper & Row Publishers, Inc. v. Nation Enterprises, 471 U.S. 539 (1985 3

Whelan Assoc. v. Jaslow Dental Laboratory, 797 F.2d 1222 (3rd Cir. 1986), *cert. denied*, 479 U.S. 1031 (1987) 1n.1,

Constitutional Provision and Statutes Cited

United States Constitution, art. I, Sec. 8, cl. 8 6

17 U.S.C. Sec. 102(a) 6

Sec. 107 6

Sec. 117 6

Other Authorities Cited

Restatement of Torts, Sec. 757, Comment b (1939) 5

New York Times, Software Jobs Go Begging, Threatening Technology Boom, p. A1. 8

POINT I

THE ISSUES PRESENTED IN THE PETITION FOR A WRIT OF CERTIORARI WERE PRESSED OR PASSED UPON BY THE COURT OF APPEALS AND ARE RIPE FOR REVIEW

This case concerns the blatant theft of computer software source code. Defendants admitted taking all of the source code for Softel's Authoring Language program. Judge Cannella found that they took this source code willfully, with total disregard for Softel's ownership rights. (App. A12, A15, A39, A40)

Softel's Authoring Language program, designed to be used to create interactive multimedia presentations, contained software developed and used by Softel in all of its commercial retail products. Defendants admitted copying the Softel Authoring Language source code verbatim into its interactive products. (App. A103, A109-10, A11-12, A116) Darsee testified that he copied Softel code for touch screen interaction, videodisc interaction and image retrieval. (A103) Defendants' expert admitted that all of Softel's Assembler routines -- some 60 pages of the source code in the Softel copyright registration stolen and used by Dragon-- were pre-existing and used in Softel's commercial Videogram product. (App. A118-19)

Defendants did not claim independent creation of their interactive programs. They asserted that they had a right to copy Softel's code because the source code of the entire program (over 3,000 lines of source code) was uncopyrightable expression.[1] (*Id.* A24)

This case is not about non-literal copying nor is it about voluntarily disclosing and sharing the source code for the Authoring language. Judge Cannella specifically found that Softel had gone to "extraordinary lengths ... to protect the privacy of [its] code" (A-28) and had consistently refused to disclose the source code to

---

[1] Defendants also argued a right to copy the code by claiming "joint work" and "fair use". Judge Cannella rejected these defenses. Regarding the "joint work" claim, Judge Cannella held that "the computer programmer is the sole author of the source and object code and the design of the system" (*Id.* A23) citing Whelan Associates v. Jaslow Dental Laboratory, Inc. 609 F. supp. 1307, 1318. (E.D. Pa. 1985) aff'd., 797 F.2d 1211 (3rd Cir. 1986) and that it was Softel who was "the author — the one who translated the "idea" into a fixed tangible expression entitled to copyright protection" (Id.)

defendants. (*Id.* A11, A12, A22, A40)

Given the admitted access and copying of the source code for virtually all of the Authoring Language, not just the "image retrieval routines", the question is how did Judge Canella conclude that Softel's source code (except for the "image retrieval routines" vaguely defined as five assembler routines) was uncopyrightable expression which defendants were permitted to copy?

Softel met the traditional criteria for copyrightability as enunciated in Feist Publications, Inc. v. Rural Telephone Service Co., 499 U.S. 340 (1991). Defendants produced no evidence that the actual source code expression of Softel's program was copied from another source. Nor did they assert that the thousands of lines of stolen code lacked the "minimum spark of creativity". Defendants offered no documentary evidence or opinion that the thousands of lines of source code for the Authoring Language were the only expression available to implement the functions of the program.

How then did Judge Canella find that the source code for Softel's[2] Authoring Language program was uncopyrightable expression? He arrived at this conclusion by applying the Altai "filtering" criteria promulgated by the Second Circuit. Judge Cannella noted the design elements of Softel's Authoring Language that were carried over into the Darsee created post-litigation programs: (1) external file structure; (2) English language commands; (3) subdividing particular code into functional modules; (4) a hierarchical series of menus with a touch screen. He applied the Altai "filtering" criteria to each of these four elements. He concluded that the idea of using menus, external files, English language commands and the modular sub-division of the computer program were common ideas. Judge Cannella determined that he did not have to examine the actual source code which implemented the files, menus, commands, and program structure, but could deny copyright protection to all of the underlying code based upon the Altai "filtration" criteria. (App. A33)

This bizarre notion, that the court can ignore the actual expression, permeated the manner in which Judge Cannella conducted the trial. Judge Cannella did not request a court-

---

[2] Computer Associates Int'l, Inc. v. Altai, Inc., 982 F.2d 693 (2nd Cir. 1992) was decided seven days before Judge Cannella issued his decision.

appointed expert pursuant to Rule 706 of the Federal Rules of Evidence to assist him in understanding the design and structure of the computer program. During the three week liability trial, he did not ask a single question on the technical issues. He barred Softel from calling a renowned expert who wrote his Doctorate on Authoring Languages and served as the head of the Association of Computer Machinery Special Interest Section on Computer Graphics in order to rebut Dragon's expert testimony. (A113) He relied entirely on the opinion testimony of Dragon's expert to determine the legal issues of copyrightability and infringement without ever posing questions to that expert. (A20)

The Altai "filtering" criteria, as demonstrated by Judge Cannella's decision, permits unauthorized copying of original fixed expression in the form of source code and the organization of that code. These "filtering" criteria, when applied to deny copyrightability to original fixed expression, are clearly wrong and inconsistent with the U.S. Supreme Court rulings in Feist and Harper & Row Publishers, Inc. v. Nation Enterprises, 471 U.S. 539 (1985), the clear language of the Copyright statute, and Congress's expressed intent to protect original expression embodied in a computer program.

Softel raised the issue of Judge Cannella's failure to accord copyright and trade secret protection to 70% of the literal code of its Authoring Language program with the Court of Appeals. Under "Issues Presented for Review", Softel asked whether the court committed error in holding that Softel's Authoring Language computer program source code was not copyrightable expression or a protected trade secret except for certain image retrieval routines within it. Softel explained to the Court of Appeals that it claimed copyright and trade secret protection for all of the specific lines of code (not just five assembler routines), and the structure of the program expressed in the source code. (Softel's Brief on Appeal at p. 10) Defendants admitted studying that structure and copying it.

Softel objected to the purported "filtration" of original source code expression based upon the notion that if the "idea" is "common" or based upon some expert's notion of "common programming techniques" or "industry, mechanical or software standards" then the expression implementing those ideas is uncopyrightable. (A-76)

The Second Circuit Court of Appeals affirmed Judge Cannella's extension of the Altai "filtering" criteria to the context of admitted

literal copying of source code. It affirmed Judge Cannella's extinguishing copyright and trade secret protection for the source code.

The Court of Appeals remanded only on the narrow issue of whether the combination of the design elements and the manner in which they interact was protectable copyright expression. It left standing the lower court's extinguishment of copyright and trade secret protection for over 70% of the original source code defendants stole and used. The Court of Appeals thus approved and passed upon the issue of whether the novel Altai "filtering" criteria could be used to permit copying of thousands of lines of original computer source code and the manner in which that code is organized into modules and sub-programs.

The issues presented in the petition for a writ of certiorari are ripe for review. The Second Circuit's ruling on uncopyrightablility and trade secret unprotectability of original expression in source code based upon the improper "Altai" filtering criteria is final and not within the remand.

POINT II

THIS CASE CONCERNS ISSUES OF FUNDAMENTAL IMPORTANCE TO THE SOFTWARE INDUSTRY.

The implications of the Second Circuit's ruling in this case for the software industry are profound. If the copyrightability of the fixed source code expression implementing a program's design elements can be denied if the design element is deemed "common", then copyright protection for source code does not exist. For example, virtually every computer program uses files. Since using files is a "common programming technique", under the Second Circuit's ruling, all of the source code creating, manipulating and executing those files is uncopyrightable, no matter how it is written, and unauthorized use of the code is permissible.

For another example, take the pervasive "cash" machine. All of these machines use a menu and either a touch screen or buttons where the user indicates his choice: i.e. cash withdrawal, deposit, account inquiry. Under the Second Circuit ruling, the computer code that implements the user's choice is uncopyrightable since the use of menus and choices for banking transactions via "cash" machines is "common" and dictated by industry standards of logic and efficiency. Thus, if an employee of the software developer for such software steals his employer's computer code and delivers it to a competitor to develop a competing program, the stolen code

can be copied with impunity because it implements the common idea of using "hierarchal menus" and touch screen input.

The Abstractions-Filtration-Test was initially promulgated as an alternative analysis for determining copying as a factual matter where defendants claimed independent creation and there was no literal copying. However, as both Altai and this case demonstrate, if the court "filters out" all of the copied elements, whether it is code or the program's organization, then it never has to reach the comparison step or the legal conclusion of whether the copying was an illicit appropriation. Thus, what was initially offered as a means of determining the structure of a copyrighted work so that it could be compared has evolved into a means of extinguishing the author's property interest in its computer program at every level — from the literal code, to the organization of that code, to the combination and interaction of the code modules and sub-programs with the result that it never has to be compared.

When the Altai decision was released, the software industry understood that it would be much harder to prove copyright infringement where the infringer had access to the source code and thus the opportunity to alter the code so that there was no literal similarity. The fall back position for the industry was state trade secret law which had always protected the secret source code of the program and the information or ideas embodied in that code. Although trade secret is governed by state law (New York follows the Restatement of Torts definition as do other jurisdictions that do not have a trade secret statute) certain fundamental principles are inherent in all those laws. Trade secret universally offers broader protection, including ideas, processes, methods and any secret information which provides a competitive business advantage to the owner. It covers any commercial use of the secret, not just copying. It also enforces a fundamentally different public policy: ethical business conduct among competitors.

The Altai decision, as applied by Judge Cannella, "filters" out source code from trade secret protection.

Congress recognized the importance of maintaining dual protection of computer programs and specifically provided that copyright does not pre-empt trade secret law. 17 U.S. C. 301(b). In this case, the District Courts crafted a parallel trade secret and copyright analysis by applying the Altai filtering criteria and copyright damage formulas to the purported trade secret analysis. Softel argued in its brief on appeal that "Anything less than an award of the

value of the trade secret [source code] deprives plaintiff of a remedy under the trade secret laws and effectively preempts the trade secret claim. (Softel's Brief at p. 43) Although the Second Circuit avoided this issue in its discussion, it implicitly passed upon the issue by affirming the District Courts' rulings .

The United States Supreme Court has the power to insure that the copyright statute's interface with state law is properly interpreted and enforced by the federal courts. Respondents' representation that "Under settled New York law, to be protectable the claimed trade secret must be 'novel" is plain error. Indeed, the Second Circuit in this case said that novelty in the patent sense is never an element of trade secret protection. (App. A81) Respondents' citation to cases concerning "theft of an idea" are irrelevant. Darsee admitted that he had access to the source code for the entire computer program and the library of assembly language routines (*Id.* A12) all of which was obtained by "improper means" (*Id.* A35). The District Court found that Softel went to extraordinary lengths to protect the privacy of the code (*Id.* A28) .

Darsee did not steal an "idea" for an Authoring Language computer program and he did not create his interactive programs based on mere discussions of how to operate the program. He stole 101 pages of computer code for a complete, working, commercially valuable program and copied nearly all of it into his earliest programs.

The right to enlarge or contract the scope of copyright protection remains exclusively the prerogative of Congress and only Congress. See U.S. Const., art. I, Sec. 8, cl. 8. The Copyright Act of 1976 was a comprehensive revision of the Copyright Act of 1909. Congress spent several decades studying the issues. Congress codified the narrow confines of "permissible" copying of original fixed expression protected under 17 U.S.C. 102(a) in the "Fair Use" statute, 17 U.S.C. 107 and it amended 17 U.S.C. 117 in 1980 to permit the holder of an authorized copy of a computer program to make one archival copy.

Respondents argue that Softel presents a broad challenge to Altai. However, Altai created a broad challenge to copyright protection for computer software. Where the court wants to find copyright protection, it does so by applying the traditional "originality" and idea/expression merger doctrine. Where it wishes to deny copyrightability, its applies the Altai "filtering" criteria. This double standard is evident in this case. Judge Cannella applied the

traditional idea/merger test to find that Softel's "Image Retrieval Routines" were protected because they were not the only way to express that "idea". (A24-25). For the remainder of the copied code,he applied the Altai criteria and found no copyright protection.

The Eleventh Circuit adopted Judge Cannella's understanding that Altai enlarged the scope of "permissible" copying. In Bateman v. Mnemonics, 79 F.3d 1532 (11th Cir. 1996), the Eleventh Circuit held that the Altai "filtering" criteria must be applied to determine the copyrightablility of source code that defendants admitted copying. The Court explained that to escape liability the alleged infringer had only to prove that the copied portion of the copyrighted work was "unworthy" of copyright protection *Id.* at 1541 and that the Altai filtering criteria could be used to establish "unworthiness". *Id.* at n. 21. In adopting the Altai "filtering" criteria to eliminate copyright protection for literal code, the Eleventh Circuit acknowledged the disagreement among the Circuits on this issue, stating:

> We recognize that the Altai test was formulated by the Second Circuit to help "determine the scope of copyright protection that extends to a computer program's non-literal structure." (Citations omitted) In fact, other circuit courts disagree on whether its application should be limited to instances of nonliteral copying, or whether it is equally applicable to cases involving literal copying. (*Id.* at 1545)

Respondents' representation in their Brief in Opposition that there is uniformity among the Circuits is belied by the Eleventh Circuit's own words. The First and Third Circuits reject Altai "filtering" criteria to determine literal copying and, among those Circuits that profess to adopt the A-F-C test, there are inconsistencies in the manner in which they determine whether or not a particular expression is "filtered" out.

If the trivialization of software development instigated by the Second Circuit becomes the prevalent rule, then the will of Congress and the statutory scheme it enacted will be thwarted. Contrary to the Second Circuit's representation, destruction of intellectual property rights in and to the original expression in computer software does not prevent monopoly. It encourages it. Why would an entrepreneur invest hundreds of thousands of dollars or millions of dollars to develop innovative products if the computer code can be stolen and used by competitors with impunity? Why would parents encourage their child to obtain a quality education in

computer science if the law says that their intellectual efforts in designing a complete working computer program is mere labor, the intellectual equivalent of organizing a list of names alphbetically in a telephone directory? See Altai at 982 F.2d 693 at 711 analogizing to facts in Feist; See also New York Times, January 13, 1998, at p. A1 "Software Jobs Go Begging, Threatening Technology Boom"); and why would this country's "best" and "brightest" pursue a career in the software industry if the courts of this country perceive their efforts unworthy of protection?

The issues presented in the Petition for a Writ of Certiorari are of fundamental importance to intellectual property protection for computer software and require guidance and clarification from this Court.

CONCLUSION

The petition for a writ of Certiorari should be granted.

Respectfully submitted,

Charla R. Bikman
Counsel for Petitioner
25 Maple Lane
East Hampton, New York
(516) 324-1306